

**TEST PROCEDURES
for
HIGH LEVEL ARCHITECTURE
OBJECT MODEL TEMPLATE**

**13 May 1997
VERSION 1.1**

Prepared by:

Margaret L. Loper
Georgia Tech Research Institute
Georgia Institute of Technology
margaret.loper@gtri.gatech.edu

TABLE OF CONTENTS

| | |
|-------------------------------------|---|
| 1. OVERVIEW | 3 |
| 1.1 DEFINITIONS | 3 |
| 1.2 ORGANIZATION OF TEST PROCEDURES | 3 |

| | |
|-------------------------------|----|
| 1.3 TEST METHODS | 3 |
| 2. CLASS STRUCTURES | 3 |
| 2.1 Traceability | 4 |
| 2.2 SOM Testing | 4 |
| 2.3 FOM Testing | 4 |
| 3. OBJECT INTERACTIONS | 5 |
| 3.1 Traceability | 5 |
| 3.2 SOM Testing | 5 |
| 3.3 FOM Testing | 6 |
| 4. ATTRIBUTES AND PARAMETERS | 6 |
| 4.1 Traceability | 7 |
| 4.2 Attribute/Parameter Table | 7 |
| 4.2.1 SOM Testing | 7 |
| 4.2.2 FOM Testing | 8 |
| 4.3 Enumerated Datatype Table | 9 |
| 4.3.1 SOM Testing | 9 |
| 4.3.2 FOM Testing | 9 |
| 4.4 Complex Datatype Table | 9 |
| 4.4.1 SOM Testing | 9 |
| 4.4.2 FOM Testing | 10 |
| 5. LEXICON | 11 |
| 5.1 Traceability | 11 |
| 5.2 SOM Testing | 11 |
| 5.3 FOM Testing | 11 |
| 6. COMPONENT STRUCTURES | 11 |
| 6.1 Traceability | 12 |
| 6.2 SOM Testing | 12 |
| 6.3 FOM Testing | 12 |
| 7. ASSOCIATIONS | 13 |
| 7.1 Traceability | 13 |
| 7.2 SOM Testing | 13 |
| 7.3 FOM Testing | 13 |
| 8. OBJECT MODEL METADATA | 14 |
| 8.1 Traceability | 14 |
| 8.2 SOM Testing | 14 |
| 8.3 FOM Testing | 15 |

1. OVERVIEW

This document contains test procedures for the High Level Architecture Object Model Template (OMT), v1.1 [1] and the Object Model Template Extensions, v1.0 [2].

1.1 DEFINITIONS

The following definitions apply through this document:

FUT The Federate or Federation Under Test.

1.2 ORGANIZATION OF TEST PROCEDURES

The OMT Test Procedures are organized into two parts with the following sections, Part I: Object Class Structures, Object Interactions, Attributes and Parameters, Enumerated Data Types, Lexicon and Part II: Component Structures, Associations, Object Model Metadata. Part I tracks with the tables from the OMT [1] document and Part II tracks with the tables in the OMT Extensions [2] document.

The OMT test procedures define the tables that are required for both Simulation Object Models (SOMs) and Federation Object Models (FOMs), and in some instances, the allowed set of information which can be used to populate those tables. For each table, the parameters are specified as one of three values: user specified, reference to a section in these test procedures, or a specific value. Additionally, some parameters are specified as optional. This indicates that the parameter can be optionally specified in the table.

1.3 TEST METHODS

Since the process for developing SOMs and FOMs is now a manual process, testing is also manual. This consists of a person verifying that there is consistency across tables entries and verifying that appropriate information is entered into SOM and FOM tables. Currently, OMT development tools are being prototyped to produce SOMs and FOMs. It is expected that these tools or their output will be tested to verify conformance with the OMT, thus facilitating automated testing.

PART I: OMT

2. CLASS STRUCTURES

A HLA object model class describes a collection of objects with some properties, behavior, relationships, and semantics in common. A class hierarchy for a HLA object model is a structure of class-subclass relations between classes of objects from the simulation or federation domain.

2.1 Traceability

Section 3.1 Object Class Structure Table [1]

2.2 SOM Testing

The Object Class Structure Table is mandatory for all Federates. A SOM must contain at least one object class.

The FUT shall have an OMT entry containing the following information for each object class or subclass:

| | |
|---------------------------|------------------------------|
| Object Class Name: | user specified |
| Publishable/Subscribable: | P, S, PS, N (in parentheses) |

All class-subclass relationships between object classes shall be documented via the Object Class Structure Table.

All object class names shall be globally unique.

Any object class shall have at most one superclass.

Any object class referenced in any other component (table) of a HLA SOM shall be included in the Object Class Structure Table.

Any object class with publishable attributes shall be included in the Object Class Structure Table.

2.3 FOM Testing

The Object Class Structure Table is mandatory for all Federations. A FOM must contain at least one object class.

The FUT shall have an OMT entry containing the following information for each object class or subclass:

| | |
|---------------------------|---------------------------|
| Object Class Name: | user specified |
| Publishable/Subscribable: | S, PS, N (in parentheses) |

All class-subclass relationships between object classes shall be documented via the Object Class Structure Table.

All object class names shall be globally unique.

Any object class shall have at most one superclass.

Any object class referenced in any other component (table) of a HLA FOM shall be included in the Object Class Structure Table.

Any object class with publishable attributes shall be included in the Object Class Structure Table.

3. OBJECT INTERACTIONS

An interaction is an explicit action taken by an object, that can optionally be directed toward another object, geographical area, etc. Interactions are specified in the object interaction table of HLA object models in terms of the interaction structure, the classes of the initiating and receiving objects and their affected attributes, and the parameters of the interaction.

3.1 Traceability

Section 3.2 Object Interaction Table [1]

3.2 SOM Testing

The Object Interaction table is mandatory for Federates that support interactions.

The FUT shall have an OMT entry containing the following information for each abstract interaction class:

| | |
|-------------------------|--|
| Interaction Class Name: | user specified |
| Hierarchical Structure: | user specified (dot notation)/optional |

The FUT shall have an OMT entry containing the following information for each concrete interaction class:

| | |
|--|---|
| Interaction Class Name: | user specified |
| Initiating Object Class Name(s): | selected from table in section 2.2 |
| Initiating Object Affected Attribute(s): | selected from table in section 4.2.1 /optional |
| Affected Attribute Comment Field: | user specified/optional |
| Receiving Object Class Name(s): | selected from table in section 2.2 /optional |
| Receiving Object Affected Attribute(s): | selected from table in section 4.2.1 /optional |
| Affected Attribute Comment Field: | user specified/optional |
| Interaction Parameter(s): | selected from table in section 4.2.1 /optional |
| Init/Sense/React: | I, S, R, IS, IR |

All class-subclass relationships between interaction classes shall be documented via the Object Interaction Table.

All interaction class names shall be globally unique.

Any interaction class shall have at most one superclass.

3.3 FOM Testing

The Object Interaction table is mandatory for Federations supporting interactions.

The FUT shall have an OMT entry containing the following information for each abstract interaction class:

| | |
|-------------------------|--|
| Interaction Class Name: | user specified |
| Hierarchical Structure: | user specified (dot notation)/optional |

The FUT shall have an OMT entry containing the following information for each concrete interaction class:

| | |
|--|---|
| Interaction Class Name: | user specified |
| Initiating Object Class Name(s): | selected from table in section 2.3 |
| Initiating Object Affected Attribute(s): | selected from table in section 4.2.2 /optional |
| Affected Attribute Comment Field: | user specified/optional |
| Receiving Object Class Name(s): | selected from table in section 2.3 /optional |
| Receiving Object Affected Attribute(s): | selected from table in section 4.2.2 /optional |
| Affected Attribute Comment Field: | user specified/optional |
| Interaction Parameter(s): | selected from table in section 4.2.2 /optional |
| Init/Sense/React: | IR |

All class-subclass relationships between interaction classes shall be documented via the Object Interaction Table.

All interaction class names shall be globally unique.

Any interaction class shall have at most one superclass.

4. ATTRIBUTES AND PARAMETERS

Each class of simulation domain objects is characterized by a fixed set of attribute types. These attributes are named portions of their object's state whose values can change over time. Public attributes are those domain object attributes whose values may be published through the RTI and provided to other federates in a federation. Parameters provide the

information required for recipients of interactions to calculate the appropriate effects of that interaction.

4.1 Traceability

Section 3.3 Attribute/Parameter Table [1]

4.2 Attribute/Parameter Table

4.2.1 SOM Testing

The Attribute/Parameter table is mandatory for Federates that exchange data via attribute updates and/or through associating parameters with interactions.

The FUT shall have an OMT entry containing the following information for each object or interaction class specified:

| | |
|--------------------------------|--|
| Object/Interaction Class Name: | selected from table in section 2.2/3.2 |
|--------------------------------|--|

The FUT shall have an OMT entry containing the following information for each object attribute or interaction parameter specified:

| | |
|--------------------------|---|
| Attribute/Parameter: | user specified |
| Datatype: | RTI basetype or user defined |
| Cardinality: | user specified |
| Units: | user specified or N/A |
| Resolution: | user specified or N/A |
| Accuracy: | user specified |
| Accuracy Condition: | user specified or N/A (Accuracy = perfect -> Accur Cond = always) |
| Update Type: | static, periodic, conditional or N/A |
| Update Condition: | user specified or N/A |
| Transferable/Acceptable: | T, A, TA, N |
| Updateable/Reflectable: | U, R, UR |

All of the public attributes associated with each object class shall be included in the Attribute/Parameter Table.

All of the parameters associated with each interaction class shall be included in the Attribute/Parameter Table.

If the specified datatype is user defined, the identifier shall be included and characterized in either the Enumerated Datatype Table or the Complex Datatype Table.

If the specified datatype is user defined, the entry for “Units”, “Resolution”, “Accuracy Condition” shall be “N/A”.

If the Attribute/Parameter Table entry is a parameter, the entry for “Update Type”, “Update Condition”, “Transferable/Acceptable”, and “Updateable/Reflectable” shall be “N/A”.

4.2.2 FOM Testing

The Attribute/Parameter table is mandatory for Federations that exchange data via attribute updates and/or through associating parameters with interactions.

The FUT shall have an OMT entry containing the following information for each object or interaction class specified:

| | |
|--------------------------------|--|
| Object/Interaction Class Name: | selected from table in section 2.4/3.4 |
|--------------------------------|--|

The FUT shall have an OMT entry containing the following information for each object attribute or interaction parameter specified:

| | |
|--------------------------|---|
| Attribute/Parameter: | user specified |
| Datatype: | RTI basetype or user defined |
| Cardinality: | user specified |
| Units: | user specified or N/A |
| Resolution: | user specified or N/A |
| Accuracy: | user specified |
| Accuracy Condition: | user specified or N/A (Accuracy = perfect -> Accur Cond = always) |
| Update Type: | static, periodic, conditional or N/A |
| Update Condition: | user specified or N/A |
| Transferable/Acceptable: | TA, N |
| Updateable/Reflectable: | UR |

All of the public attributes associated with each object class shall be included in the Attribute/Parameter Table.

All of the parameters associated with each interaction class shall be included in the Attribute/Parameter Table.

If the specified datatype is user defined, the identifier shall be included and characterized in either the Enumerated Datatype Table or the Complex Datatype Table.

If the specified datatype is user defined, the entry for “Units”, “Resolution”, “Accuracy Condition” shall be “N/A”.

If the Attribute/Parameter Table entry is a parameter, the entry for “Update Type”, “Update Condition”, “Transferable/Acceptable”, and “Updateable/Reflectable” shall be “N/A”.

4.3 Enumerated Datatype Table

4.3.1 SOM Testing

If a user defined enumerated datatype is specified in section 4.2.1, this table is mandatory.

The FUT shall have an OMT entry containing the following information for each enumerated datatype:

| | |
|-----------------|--|
| Enumeration ID: | user defined datatype from section 4.2.1 |
| Enumerator: | user specified |
| Representation: | user specified |

4.3.2 FOM Testing

If a user defined enumerated datatype is specified in section 4.2.2, this table is mandatory.

The FUT shall have an OMT entry containing the following information for each enumerated datatype:

| | |
|-----------------|--|
| Enumeration ID: | user defined datatype from section 4.2.2 |
| Enumerator: | user specified |
| Representation: | user specified |

4.4 Complex Datatype Table

4.4.1 SOM Testing

If a user defined complex datatype is specified in section 4.2.1, this table is mandatory.

The FUT shall have an OMT entry containing the following information for each complex datatype specified:

| | |
|------------------------------|--|
| Complex Datatype Identifier: | user defined datatype from section 4.2.1 |
|------------------------------|--|

The FUT shall have an OMT entry containing the following information for each individual field in the complex datatype:

| | |
|--------------|-------------------------------|
| Field Name: | user specified |
| Datatype: | RTI basedtype or user defined |
| Cardinality: | user specified |
| Units: | user specified or N/A |

| | |
|---------------------|---|
| Resolution: | user specified or N/A |
| Accuracy: | user specified or N/A |
| Accuracy Condition: | user specified or N/A (Accuracy = perfect -> Accur Cond = always) |

If the datatype for a field is user defined, the datatype identifier shall be included and characterized in either the Enumerated Datatype Table or elsewhere in the Complex Datatype Table.

If the datatype for a field is user defined, the entry for “Units”, “Resolution”, “Accuracy”, and “Accuracy Condition” shall be “N/A”.

4.4.2 FOM Testing

If a user defined complex datatype is specified in section 4.2.2, this table is mandatory.

The FUT shall have an OMT entry containing the following information for each complex datatype specified:

| | |
|------------------------------|--|
| Complex Datatype Identifier: | user defined datatype from section 4.2.2 |
|------------------------------|--|

The FUT shall have an OMT entry containing the following information for each individual field in the complex datatype:

| | |
|---------------------|---|
| Field Name: | user specified |
| Datatype: | RTI basedtype or user defined |
| Cardinality: | user specified |
| Units: | user specified or N/A |
| Resolution: | user specified or N/A |
| Accuracy: | user specified or N/A |
| Accuracy Condition: | user specified or N/A (Accuracy = perfect -> Accur Cond = always) |

If the datatype for a field is user defined, the datatype identifier shall be included and characterized in either the Enumerated Datatype Table or elsewhere in the Complex Datatype Table.

If the datatype for a field is user defined, the entry for “Units”, “Resolution”, “Accuracy”, and “Accuracy Condition” shall be “N/A”.

5. LEXICON

In addition to the specification of which particular classes of data will be exchanged among the members of a HLA federation to meet a given set of federation requirements, it is also imperative that the federates achieve a common understanding of the semantics of the data being exchanged if interoperability between simulations is to be attained. The Lexicon provides a means for federations to document the definitions of all terms utilized during construction of FOMs, and for individual federates to document the definitions of all terms provided in their SOMs.

5.1 Traceability

Section 4.0 SOM/FOM Lexicon [1]

5.2 SOM Testing

The Lexicon table is mandatory for all federates. The terms listed in the first column of each table specified in sections 2.2, 3.2, 4.2.1, 4.3.1, and 4.4.1 must be defined in the lexicon.

The FUT shall have an OMT entry containing the following information from each Object Class Structure, Object Interaction, and Attribute/Parameter Table:

| | |
|----------------------|--|
| Object Class: | defined in section 2.2 |
| Object Interaction: | defined in section 3.2 |
| Attribute/Parameter: | defined in section 4.2.1, 4.3.1, 4.4.1 |

5.3 FOM Testing

The Lexicon table is mandatory for all federations. The terms listed in the first column of each table specified in sections 2.3, 3.3, 4.2.2, 4.3.2, and 4.4.2 must be defined in the lexicon.

The FUT shall have an OMT entry containing the following information from each Object Class Structure, Object Interaction, and Attribute/Parameter Table:

| | |
|----------------------|--|
| Object Class: | defined in section 2.3 |
| Object Interaction: | defined in section 3.3 |
| Attribute/Parameter: | defined in section 4.2.2, 4.3.2, 4.4.2 |

PART II: OMT Extensions

6. COMPONENT STRUCTURES

A component structure for a HLA object model is a structure of part-whole relations between classes of objects from the simulation or federation domain. A part-whole

relation between two classes indicates that objects from one class are parts (or components) of composite objects from another class. A HLA component structure is simply a set of these part-whole (or component) relations along with their cardinalities. Typical part-whole relations include those between the assemblies and subassemblies of machines, as well as inclusion relations between parts of an organization, such as the Divisions in a Corps or the Brigades in a Division.

6.1 Traceability

Section 3.1 Component/Structure Table [2]

6.2 SOM Testing

This table is optional for federates.

The FUT shall have an OMT entry containing the following information for each composite object class included in the Component Structure Table:

| | |
|---------------|----------------|
| Object Class: | user specified |
|---------------|----------------|

The FUT shall have an OMT entry containing the following information for each object class designated as a component of a composite object class:

| | |
|--------------------|-------------------------|
| Object Class Name: | user specified |
| Cardinality: | user specified/optional |

All part-whole relationships between object classes shall be documented via the Component Structure Table.

6.3 FOM Testing

This table is optional for federations.

The FUT shall have an OMT entry containing the following information for each composite object class included in the Component Structure Table:

| | |
|---------------|----------------|
| Object Class: | user specified |
|---------------|----------------|

The FUT shall have an OMT entry containing the following information for each object class designated as a component of a composite object class:

| | |
|--------------------|-------------------------|
| Object Class Name: | user specified |
| Cardinality: | user specified/optional |

All part-whole relationships between object classes shall be documented via the Component Structure Table.

7. ASSOCIATIONS

Part-whole relations are just one type of association between classes. Other associations include command and control associations between military command units, as well as usage and operations relations between vehicles and stations for refueling, supply, and maintenance. The associations component of a HLA object model is designed to capture all such other associations which are important for assessments of interoperability and reuse. All such associations are indicative of a lasting relationship between objects, as opposed to the momentary relations of interaction events.

7.1 Traceability

Section 3.2 Association Table [2]

7.2 SOM Testing

This table is optional for federates.

The FUT shall have an OMT entry containing the following information for each association:

| | |
|---------------|--|
| First Class: | defined in section 2.2 |
| Cardinality: | user specified/optional |
| Role: | user specified (in parentheses) /optional |
| Association: | user specified |
| Second Class: | defined in section 2.2 |
| Cardinality: | user specified/optional |
| Role: | user specified/optional |

7.3 FOM Testing

This table is optional for federations.

The FUT shall have an OMT entry containing the following information for each association:

| | |
|--------------|-------------------------|
| First Class: | defined in section 2.3 |
| Cardinality: | user specified/optional |
| Role: | user specified/optional |

| | |
|---------------|-------------------------|
| Association: | user specified |
| Second Class: | defined in section 2.3 |
| Cardinality: | user specified/optional |
| Role: | user specified/optional |

8. OBJECT MODEL METADATA

Metadata for a HLA object model consists of information about a federation or simulation being modeled as a whole, as contrasted with the more specific information about object classes, attributes, and interactions which are recorded in the components of a FOM or SOM. For simulations, metadata includes information about general operating characteristics, including general hardware and software characteristics or constraints. Such information can prove helpful in determining the potential for interoperability with other federates and/or federations. For federations, metadata is useful for federation developers to determine the potential reuse of all or parts of an existing FOM.

8.1 Traceability

Section 3.3 Object Model Metadata [2]

8.2 SOM Testing

This table is optional for federates.

The FUT shall have an OMT entry containing the following information for each SOM:

| | |
|------------------------------------|--------------------------|
| Name: | simulation name |
| Class: | V, L, C, Hybrid/optional |
| Domain: | user specified/optional |
| Modification Date: | user specified/optional |
| Sponsor: | user specified/optional |
| Developer: | user specified/optional |
| Point of Contact: | user specified/optional |
| Security Characteristics: | user specified/optional |
| VV&A History: | user specified/optional |
| Tool Use: | user specified/optional |
| Key Features: | user specified/optional |
| Computational Capacity Capability: | user specified/optional |
| Configuration Requirements: | user specified/optional |
| Programming Languages: | user specified/optional |
| Time Management: | user specified/optional |
| Documentation References: | user specified/optional |

Other: user specified/optional

8.3 FOM Testing

This table is optional for federations.

The FUT shall have an OMT entry containing the following information for each FOM:

| | |
|---------------------------|--------------------------|
| Name: | simulation name |
| Class: | V, L, C, Hybrid/optional |
| Domain: | user specified/optional |
| Purpose: | user specified/optional |
| Modification Date: | user specified/optional |
| Sponsor: | user specified/optional |
| Developer: | user specified/optional |
| Point of Contact: | user specified/optional |
| Federation Participants: | user specified/optional |
| Security Characteristics: | user specified/optional |
| VV&A History: | user specified/optional |
| Tool Use: | user specified/optional |
| Documentation References: | user specified/optional |
| Other: | user specified/optional |

REFERENCES

- [1] DMSO, High Level Architecture Object Model Template, v1.1, 12 February 1997.
- [2] DMSO, High Level Architecture OMT Extensions, v1.0, 20 August 1996.